IST 256
Lab Week 10, Day 2 – Wednesday, November 4, 2009

1. Understanding How to Write a Procedure Call

Suppose that we have a form that lets the user figure out how much dog food to feed their dog. The form asks the user to type in the weight of their dog in pounds and how active their dog is. The button procedure gets the weight and stores it into a **Single variable called "pounds"** and stores the activity level into a **String variable called "activity"**. This form has a procedure called dogfood that will set a label to show the amount of dog food the dog should get per day. Also suppose that the form has a **label named Label3** to display the amount of dog food. Here is the procedure:

```
Public Class Form1
   ‘ This procedure takes the weight and activity of a dog,
   ‘     and displays the amount of dog food per day in a label
   Private Sub dogfood ( ByVal wt as Single, ByVal active as String, ByRef  lab as Label)
         Dim amount as Single
         If (active = “slow”) then
                  Amount = wt * 0.1
         Else
                  If (amount = “active”) then
                           Amount = wt * 0.2
                  Else
                           If (amount = “hyper”) then
                                    Amount = wt * 0.4
                           Else
                                    Amount = wt * 0.2
                           End if
                  End if
         End if
         lab.Text = “Amount Per Day = “ & Format(amount, “standard”)
   End Sub
….
End Class
```

a. Is the following a valid procedure call to dogfood, and if not, why not?

```
         Call dogfood (pound, 10, Label3)
```

b. Suppose that the value of the variable "weight" is 10.0 and the value of the variable "activity" is "hyper". After this procedure call, write what output will be given in the label:

```
         Call dogfood ( pounds, activity, Label3)
```

2. Understanding how to write a procedure

In a VB project with 3 forms, suppose that there is a module. Write a procedure in the module :

1. Decide on a procedure name
2. There is only one parameter and it is a TextBox
3. The procedure does the following:
   - If an integer value from the TextBox is between 0 to 99, then do nothing
   - If an integer value from the TextBox is smaller then 0 or larger then 99, popup a msgbox with "Mistake!", and change the TextBox value to 0

3. Understanding how to read values from a file

Write a small program to read the values from a file where each line has the name of a person and their age:

Mike, 25
Kate, 19
Xiaozhong, 30
Obama, 48

The program should compute the average age of the people. (You don't have to write a part to display the average age.)

4. Understanding a progam that writes to a file

Suppose that we have the following program to write to a file:

```
Dim birthyear as Integer
birthyear = 1979
Dim age as Integer
age = 2009 - birthyear

FileOpen(1, “report.txt”, OpenMode.Output)

PrintLine(1, “Xiaozhong")
PrintLine(1, “”)
Print(1, “Birth Year: ", birthyear, vbCrLf)
Print(1, “Age: ", age)

Closefile(1)
```

What lines of text will be written to the file?

5. Understanding an Array Program

a. Suppose that we have a file named “examples.txt” and it has 5 numbers in it:

5
9
-6
4
1

We also have a form with a button procedure that reads these numbers into an array called “examples” and uses them.

```
Dim examples ( 0 to 4 ) As Integer
Dim index As Integer

' read the example numbers from the file and store in the array
FileOpen ( 1, “examples.txt”, OpenMode.Input)
index = 0
While Not EOF(1)
      Input ( 1, examples(index))
      index = index + 1
End While

' standard array loop to double each element
For index = 0 To 4
      examples(index) = examples(index) * 2
Next
```

b. Without running the program, think about the values of the variables during the execution of the program. Write the values of the array elements after the While loop and after the For loop:

Variable	Value after While loop	Value after For loop
examples(0)		
examples(1)		
examples(2)		
examples(3)		
examples(4)		

For labs this week, do not hand in any lab papers. Show your labs to Prof. McCracken to check off that your work is complete.